

Docket No.: HI-0066

PATENT

#2
2-14-02
JC996 U.S. PRO
10/025851
12/26/01

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of :

Jeong Il BANG :

Serial No.: New U.S. Patent Application :

Filed: December 26, 2001 :

For: METHOD FOR DEBUGGING IN APPLICATION PROGRAM AND
APPARATUS THEREOF

TRANSMITTAL OF CERTIFIED PRIORITY DOCUMENT

Assistant Commissioner of Patents
Washington, D. C. 20231

Sir:

At the time the above application was filed, priority was claimed based on the
following application:

Korean Patent Application No. 83634/2000, filed December 28, 2000

A copy of each priority application listed above is enclosed.

Respectfully submitted,
FLESHNER & KIM, LLP

Carl R. Wesolowski

Daniel Y.J. Kim
Registration No. 36,186
Carl R. Wesolowski
Registration No. 40,372

P. O. Box 221200
Chantilly, Virginia 20153-1200
703 502-9440

Date: December 26, 2001

DYK/CRW:j1d

JC996 U.S. PTO
10/025851
12/26/01



별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto
is a true copy from the records of the Korean Intellectual
Property Office.

출원번호 : 특허출원 2000년 제 83634 호
Application Number PATENT-2000-0083634

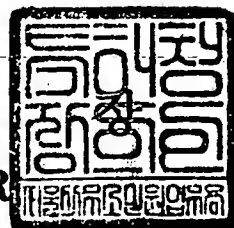
출원년월일 : 2000년 12월 28일
Date of Application DEC 28, 2000

출원인 : 엘지전자주식회사 :
Applicant(s) LG ELECTRONICS INC.



2001 년 10 월 05 일

특 허 청
COMMISSIONER



【서지사항】

【서류명】 특허출원서
【권리구분】 특허
【수신처】 특허청장
【참조번호】 0016
【제출일자】 2000.12.28.
【국제특허분류】 H04B
【발명의 명칭】 인터럽트를 이용한 응용 프로그램의 에러검출장치 및 방법.
【발명의 영문명칭】 Applied Program Bungle Detection Apparatus and Method by Interrupt
【출원인】
【명칭】 엘지전자 주식회사
【출원인코드】 1-1998-000275-8
【대리인】
【성명】 허용록
【대리인코드】 9-1998-000616-9
【포괄위임등록번호】 1999-043458-0
【발명자】
【성명의 국문표기】 방정일
【성명의 영문표기】 BANG, Jeong II
【주민등록번호】 700404-1480710
【우편번호】 431-080
【주소】 경기도 안양시 동안구 호계동 967-52
【국적】 KR
【취지】 특허법 제42조의 규정에 의하여 위와 같이 출원합니다. 대리인
허용록 (인)

【수수료】

【기본출원료】	15 면	29,000 원
【가산출원료】	0 면	0 원
【우선권주장료】	0 건	0 원
【심사청구료】	0 항	0 원
【합계】	29,000 원	

1020000083634

출력 일자: 2001/10/8

【첨부서류】

1. 요약서·명세서(도면)_1통

【요약서】

【요약】

본 발명은 응용 프로그램의 에러를 검출하는 것으로, 본 발명에 따른 인터럽트를 이용한 응용 프로그램의 에러검출 장치는, 보드에서 전체적인 제어와 처리를 하는 중앙처리장치와; 상기 중앙처리장치의 명령에 따라 수행되는 타스크가 정상적인 동작을 하는가를 검사하고, 에러검출신호를 출력하는 비교로직과; 상기 비교로직으로부터의 에러검출신호를 받아 메모리를 제어하는 메모리 제어신호를 출력하는 메모리제어기와; 상기 메모리제어기로부터의 메모리제어신호를 입력 받아, 데이터를 저장 또는 삭제하는 메모리를 포함하여 구성된다.

한편, 본 발명에 따른 인터럽트를 이용한 응용 프로그램의 에러검출 방법은, 보드가 부팅되고, 운영체제가 동작하여 수행할 타스크 아이디를 할당된 영역에 기록하는 단계; 상기 기록된 타스크 아이디에 의하여 타스크 아이디 디코더가 데이터 버스를 래치하고, 타스크 아이디 판별기가 타스크 N 신호를 발생하는 단계; 상기 타스크 아이디에 해당하는 타스크가 실행되는 단계; 상기 타스크에 따른 동작이 각각 정해진 영역에서 이루어지는가를 검색하는 단계; 상기 검색 결과 상기 타스크에 따른 동작이 정해진 영역을 벗어나서 이루어져서 그랜트 신호를 0으로 발생시켜 인터럽트가 발생하는 단계; 상기 인터럽트가 발생한 시점에 에러가 발생한 프로그램을 파악하고, 에러를 수정하는 단계를 포함하는 것을 특징으로 한다.

이 같은 본 발명에 의하면, 이제까지의 메모리 쓰기 금지를 페이지별로 하여, 온라인 상에서 프로그램의 에러가 발생했을 때 정확히 에러가 발생한 응용

프로그램과 에러발생의 이유를 알아내기가 어려웠던 것을, 응용 프로그램의 동작에 의해서 타스크별로 할당된 영역을 위반하지 않는지를 간단한 비교로직의 하드웨어적 추가만으로도 검출하여 인터럽트 신호를 발생시킴으로써 중앙처리장치에 현재 문제를 일으킨 메모리와 응용 프로그램에 대한 정보를 정확히 알려주어 에러를 처리할 수 있도록 하는 효과가 기대된다.

【대표도】

도 2

【명세서】

【발명의 명칭】

인터럽트를 이용한 응용 프로그램의 에러검출장치 및 방법.{Applied Program Bungle Detection Apparatus and Method by Interrupt}

【도면의 간단한 설명】

도 1은 종래의 메모리 보호방법에 따른 MC68060의 메모리 관리부에서 사용하는 페이지 서술자 형태를 나타낸 구조도.

도 2는 본 발명의 실시 예에 따른 인터럽트를 이용한 응용 프로그램의 에러검출장치의 구조를 나타낸 블록도.

도 3은 본 발명의 실시 예에 따른 인터럽트를 이용한 응용 프로그램의 에러검출장치의 검사부의 비교로직의 구성을 나타낸 구성도.

도 4는 본 발명의 실시 예에 따른 인터럽트를 이용한 응용 프로그램의 에러검출장치의 TASK 아이디 디코더의 구조를 나타낸 블록도.

도 5는 본 발명의 실시 예에 따른 인터럽트를 이용한 응용 프로그램의 에러검출방법의 동작순서를 나타낸 플로우 차트.

<도면의 주요부분에 대한 간단한 설명>

10 : 중앙처리장치 20 : 메모리

30 : 메모리 제어기 40 : 비교로직

41 : OR NOT 게이트 42 : AND 게이트

50 : 래치 60 : TASK 아이디 판별기

【발명의 상세한 설명】

【발명의 목적】

【발명이 속하는 기술분야 및 그 분야의 종래기술】

<11> 본 발명은 응용 프로그램의 에러를 검출하는데 있어서, 특히 인터럽트를 이용한 응용 프로그램의 에러검출방법에 관한 것이다.

<12> 종래의 응용 프로그램의 디버깅 방법으로는 메모리를 쓰기 금지하는 방법이 있었으며, 도 1은 종래의 메모리 보호방법에 따른 MC68060의 메모리 관리부에서 사용하는 페이지 서술자 형태를 나타낸 구조도이다.

<13> 도 1을 참조하면, 응용 프로그램의 에러로부터 메모리를 보호하기 위해서는 중앙처리장치 내부의 메모리 관리부(MMC; memory management unit)에서 담당하며, 예를 들어 MC68060의 메모리 관리부에서는 페이지 서술자 형태중 'W'로 표시된 것이 쓰기 금지 여부를 선택할 수 있도록 하였다. 즉 'W'가 '1'로 세팅되어 있으면, 이 페이지는 읽기만 가능하고 쓰기는 불가능했다.

<14> 즉, 임의의 프로그램에 의해 정상적으로 데이터가 저장될 메모리가 위치한 페이지만을 'W'='0'으로 세팅하여 메모리에 쓰기를 할 수 있도록 하여, 상기 임의의 프로그램이 오류를 일으켜 다른 페이지에 데이터를 쓰지 못하도록 하여 메모리를 보호하였다.

<15> 그러나, 이러한 종래의 방법은 각각의 응용 프로그램별로 사용하게 되는 메모리를 제어하는 것이 아니라, 페이지 단위로 메모리를 쓰기 금지시키는 방법이므로, 응용 프로그램마다 사용하는 스택이 다르고 가변적인 특성으로 인하여, 에

러가 검출되었을 때, 정확히 어떤 응용 프로그램에 의한 에러인지 파악이 어렵고, 응용 프로그램별로의 쓰기 금지를 할 수는 없었다.

【발명이 이루고자 하는 기술적 과제】

- <16> 상기와 같은 문제를 해결하기 위해서, 본 발명은 TASK별로 할당된 영역을 응용 프로그램의 동작에 의해서 TASK별로 할당된 영역을 위반하지 않는지를 간단한 비교로직의 하드웨어적 추가만으로도 검출하여 인터럽트 신호를 발생시킴으로써 중앙처리장치에 현재 문제를 일으킨 메모리와 응용 프로그램에 대한 정보를 정확히 알려주어 에러를 처리할 수 있도록 하는 인터럽트를 이용한 응용 프로그램의 에러검출방법을 제공함에 그 목적이 있다.

【발명의 구성 및 작용】

- <17> 본 발명에 따른 인터럽트를 이용한 응용 프로그램의 에러검출장치는,
<18> 보드에서 전체적인 제어와 처리를 하는 중앙처리장치와;
<19> 상기 중앙처리장치의 명령에 따라 수행되는 TASK가 정상적인 동작을 하는가를 검사하고, 에러검출신호를 출력하는 비교로직과;
<20> 상기 비교로직으로부터의 에러검출신호를 받아 메모리를 제어하는 메모리 제어신호를 출력하는 메모리제어기와;
<21> 상기 메모리제어기로부터의 메모리제어신호를 입력받아, 데이터를 저장 또는 삭제하는 메모리를 포함하여 구성되는 것을 특징으로 한다.

- <22> 바람직하게, 상기 비교로직은 TASK 아이디를 디코딩 해주는 TASK 아이디 디코딩부와, 각 TASK가 정해진 영역에서 이루어지는가를 검사하는 검사부를 포함하여 구성되는 것을 특징으로 한다.
- <23> 한편, 본 발명에 따른 인터럽트를 이용한 응용 프로그램의 에러검출 방법은
- <24> 보드가 부팅되고, 운영체제가 동작하여 수행할 TASK 아이디를 할당된 영역에 기록하는 단계;
- <25> 상기 기록된 TASK 아이디에 의하여 TASK 아이디 디코더가 데이터 버스를 래치하고, TASK 아이디 판별기가 TASK N 신호를 발생하는 단계;
- <26> 상기 TASK 아이디에 해당하는 TASK가 실행되는 단계;
- <27> 상기 TASK에 따른 동작이 각각 정해진 영역에서 이루어지는가를 검색하는 단계;
- <28> 상기 검색결과 상기 TASK에 따른 동작이 정해진 영역을 벗어나서 이루어져서 그랜트 신호를 0으로 발생시켜 인터럽트가 발생하는 단계;
- <29> 상기 인터럽트가 발생한 시점에 에러가 발생한 프로그램을 파악하고, 에러를 수정하는 단계를 포함하는 것을 특징으로 한다.
- <30> 바람직하게, 상기 TASK의 동작이 정해진 영역에서 이루어지는가를 검색하는 것은, 메모리에 데이터가 쓰이는 동작과 동시에 이루어지며 메모리에 데이터가 쓰이기 전에 완료되어 메모리 제어신호를 만드는 입력신호로 사용되는 것을 특징으로 한다.

- <31> 바람직하게, 상기 에러를 수정하는 것은, 에러가 발생한 TASK 아이디를 정해진 영역에 기록하여, 다시 한번 수행되도록 하는 것을 특징으로 한다.
- <32> 이하 첨부된 도면을 참조하여 자세히 설명하면 다음과 같다.
- <33> 도 2는 본 발명의 실시 예에 따른 인터럽트를 이용한 응용 프로그램의 에러 검출장치의 구조를 나타낸 블록도이고, 도 3은 본 발명의 실시 예에 따른 인터럽트를 이용한 응용 프로그램의 에러검출장치의 비교로직의 검사부의 구성을 나타낸 구성도이고, 도 4는 본 발명의 실시 예에 따른 인터럽트를 이용한 응용 프로그램의 에러검출장치의 TASK아이디 디코더의 구조를 나타낸 블록도이고, 도 5는 본 발명의 실시 예에 따른 인터럽트를 이용한 응용 프로그램의 에러검출방법의 동작순서를 나타낸 플로우 차트이다.
- <34> 도 2를 참조하면, 회로의 각 주변장치들의 동작을 제어하는 중앙처리장치(10)와, 상기 중앙처리장치(10)가 처리할 데이터를 저장하고 있는 메모리(20)와, 상기 메모리를 제어하는 메모리 제어기(30)와, 상기 메모리의 데이터에 의한 TASK가 정상적으로 수행되고 있는가를 검사하는 비교로직(40)을 포함하여 구성된다.
- <35> 여기서, 상기 비교로직(40)은 TASK 아이디를 디코딩 해주는 TASK아이디 디코딩부분과 TASK별로 정상적인 영역에서 동작하는지를 검사하는 검사부를 포함하여 구성된다.
- <36> 도 3을 참조하여, 상기 비교로직의 TASK별로 할당된 번지에 상기 중앙처리장치(10)가 쓰기를 하는지 검사하는 부분은 TASK가 수행될 때 사용되지 않는

메모리 번지의 입력을 받는 OR-NOT게이트(41)와 상기 OR-NOT게이트(41)의 출력과
타스크 N의 신호와 현재 쓰기 상태인지 읽기 상태인지를 나타내는 !WR신호를 입
력받아 그랜트 신호를 발생하는 AND게이트(42)로 구성된다.

<37> 또한 도 4를 참조하여, 상기 타스크 아이디를 디코딩 해주는 타스크 아이디
디코더는 타스크를 수행하기 위한 데이터버스를 래치 하는 래치부분(43)과, 상
기 타스크 아이디를 판별하는 타스크 아이디 판별기(44)를 포함하여 구성된다.

<38> 상기와 같은 본 발명에 따른 인터럽트를 이용한 응용 프로그램의 에러검출
방법에 대하여 첨부된 도면을 참조하여 설명하면 다음과 같다.

<39> 도 2와 도 3과 도 4와 도 5를 참조하면, 보드가 부팅하고(S501), 운영체제
가 동작하기 시작하면(S502), 상기 중앙처리장치는(10) 수행해야 할 타스크 아이
디를 정해진 영역에 기록한다(S503).

<40> 상기 기록된 타스크 아이디에 따라 타스크 아이디 디코더(40)의 래치부분
(43)은 데이터 버스를 래치 하여(S504), 다음 타스크가 실행될 때까지 유지하고,
타스크 아이디 판별기(44)는 해당하는 타스크 N의 신호를 발생한다(S505).

<41> 이때, 상기 타스크 N 신호는 비교로직의 검사부로 들어가 정상적 영역에서
동작이 이루어지는가를 검사하는데 쓰인다.

<42> 상기 단계 S505이후 해당 타스크를 수행하고(S506), 이때 각 동작이 정해진
영역에서 이루어지는가를 검색하게 된다(S507). 이때, 검사는 하는 것은, 정상적
인 영역에서 각 동작이 이루어지면, 해당 타스크의 수행에 의해 메모리가 채워지
지 않는 각 번지를 입력으로 하는 OR-NOT 게이트(41)의 출력이 1이 되고, 타스크

N신호와 함께 현재 쓰기 단계에 있으므로 !WR신호 역시 1이 되어, AND게이트의 입력이 모두 1이 되므로 그랜트 신호가 '1'이 되어 정상적인 동작이 되지만, 상기 TASK의 수행중 에러가 발생하여 쓰지 않아야 할 메모리 번지에 데이터를 쓰게 되면 OR-NOT게이트(41)로 입력되는 TASK의 번지중 어느 하나로도 '1'의 값이 입력되어 상기 OR-NOT 게이트(41)의 출력이 '0'이 되어 AND게이트의 출력이 '0'이 되므로 그랜트 값이 '0'이 되어, 에러가 발생했음을 인지하게 된다.

<43> 한편, 상기와 같은 검사를 할수 있는 검사부는 TASK의 수만큼 필요하며, 각 TASK마다 각각 에러를 검사하게 된다.

<44> 상기 단계 S507에서 상기 TASK 수행 중에 정해진 영역이외에서 동작이 이루어지는 것이 발견되면, 그랜트 신호가 '0'이 되고(S508), 인터럽트가 발생한다(S510).

<45> 상기와 같이 인터럽트가 발생되면, 비교로직(40)은 메모리 제어신호를 만드는 입력신호를 메모리 제어기(30)에 보내어, 메모리 제어기(30)에서 메모리 제어신호를 발생한다(S511).

<46> 상기 메모리 제어신호가 발생되면, 중앙처리장치(10)는 현재 에러가 발생한 응용 프로그램과 메모리 번지를 파악하고(S512), 에러가 발생한 TASK의 아이디를 정해진 영역에 기록하여, 다시 수행할 수 있도록 하여 에러를 수정한다(S513).

【발명의 효과】

<47> 이상에서 설명한 바와 같이, 본 발명에 따른 인터럽트를 이용한 응용 프로그램의 에러검출 장치 및 방법은, 이제까지의 메모리 쓰기 금지를 페이지별로 하여, 온라인 상에서 프로그램의 에러가 발생했을 때 정확히 에러가 발생한 응용 프로그램과 에러발생의 이유를 알아내기가 어려웠던 것을, 응용 프로그램의 동작에 의해서 타스크별로 할당된 영역을 위반하지 않는지를 간단한 비교로직의 하드웨어적 추가만으로도 검출하여 인터럽트 신호를 발생시킴으로써 중앙처리장치에 현재 문제를 일으킨 메모리와 응용 프로그램에 대한 정보를 정확히 알려주어 에러를 처리할 수 있도록 하는 효과가 있다.

【특허청구범위】**【청구항 1】**

보드에서 전체적인 제어와 처리를 하는 중앙처리장치와;

상기 중앙처리장치의 명령에 따라 수행되는 TASK가 정상적인 동작을 하는가를 검사하고, 에러검출신호를 출력하는 비교로직과;

상기 비교로직으로부터의 에러검출신호를 받아 메모리를 제어하는 메모리 제어신호를 출력하는 메모리제어기와;

상기 메모리제어기로부터의 메모리제어신호를 입력받아, 데이터를 저장 또는 삭제하는 메모리를 포함하여 구성되는 것을 특징으로 하는 인터럽트를 이용한 응용 프로그램의 에러검출장치.

【청구항 2】

제 1항에 있어서, 상기 비교로직은 TASK 아이디를 디코딩 해주는 TASK 아이디 디코딩부와, 각 TASK가 정해진 영역에서 이루어지는가를 검사하는 검사부를 포함하여 구성되는 것을 특징으로 하는 인터럽트를 이용한 응용 프로그램의 에러검출 장치.

【청구항 3】

보드가 부팅되고, 운영체제가 동작하여 수행할 TASK 아이디를 할당된 영역에 기록하는 단계;

상기 기록된 TASK 아이디에 의하여 TASK 아이디 디코더가 데이터 버스를 래치하고, TASK 아이디 판별기가 TASK N 신호를 발생하는 단계;

상기 태스크 아이디어에 해당하는 태스크가 실행되는 단계;

상기 태스크에 따른 동작이 각각 정해진 영역에서 이루어지는가를 검색하는 단계;

상기 검색결과 상기 태스크에 따른 동작이 정해진 영역을 벗어나서 이루어져서 그랜트 신호를 0으로 발생시켜 인터럽트가 발생하는 단계;

상기 인터럽트가 발생한 시점에 에러가 발생한 프로그램을 파악하고, 에러를 수정하는 단계를 포함하는 것을 특징으로 하는 인터럽트를 이용한 응용 프로그램의 에러검출 방법.

【청구항 4】

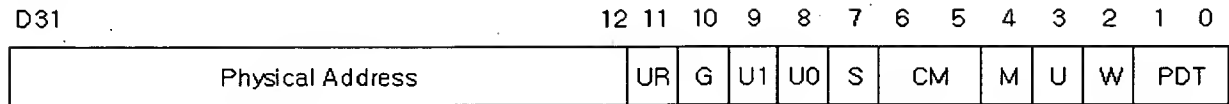
제 3항에 있어서, 상기 태스크의 동작이 정해진 영역에서 이루어지는가를 검색하는 것은, 메모리에 데이터가 써지는 동작과 동시에 이루어지며 메모리에 데이터가 써지기 전에 완료되어 메모리 제어신호를 만드는 입력신호로 사용되는 것을 특징으로 하는 인터럽트를 이용한 응용 프로그램의 에러검출 방법.

【청구항 5】

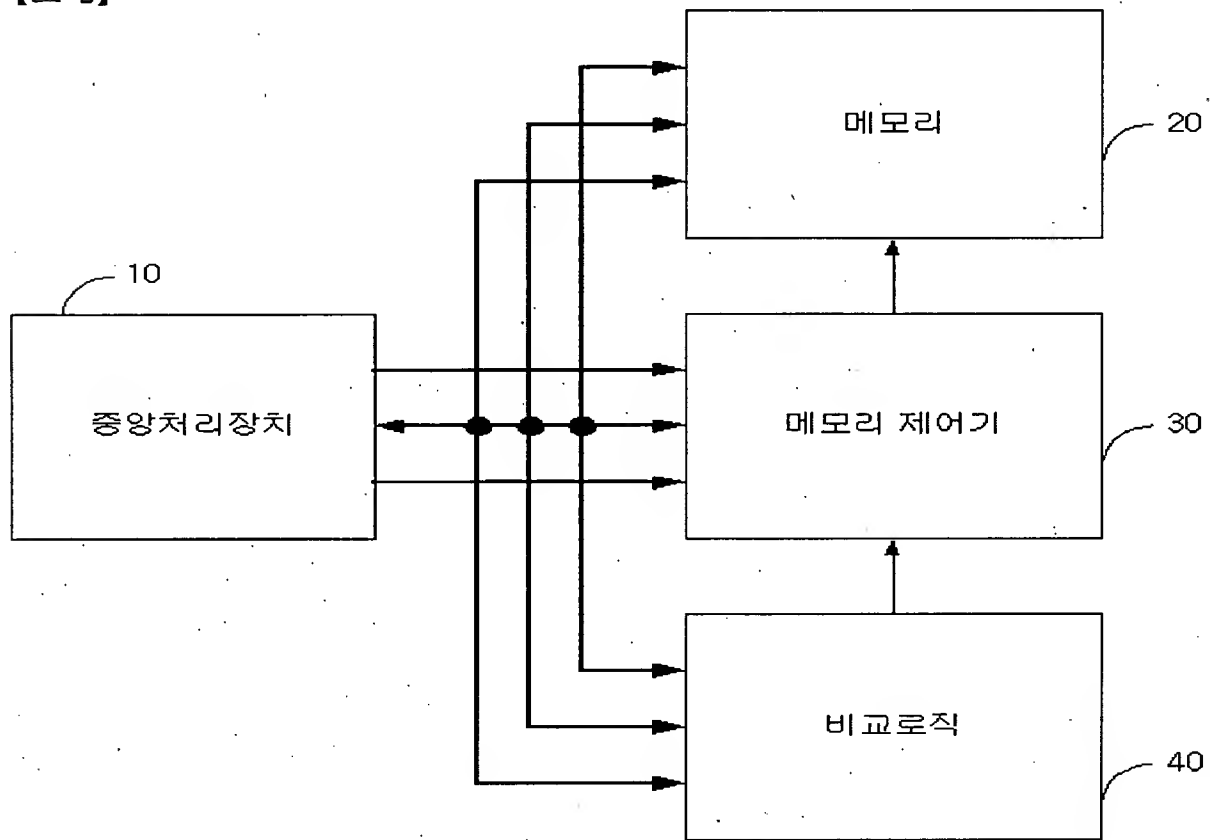
제 3항에 있어서, 상기 에러를 수정하는 것은, 에러가 발생한 태스크 아이디어를 정해진 영역에 기록하여, 다시 한번 수행되도록 하는 것을 특징으로 하는 인터럽트를 이용한 응용 프로그램의 에러검출 방법.

【도면】

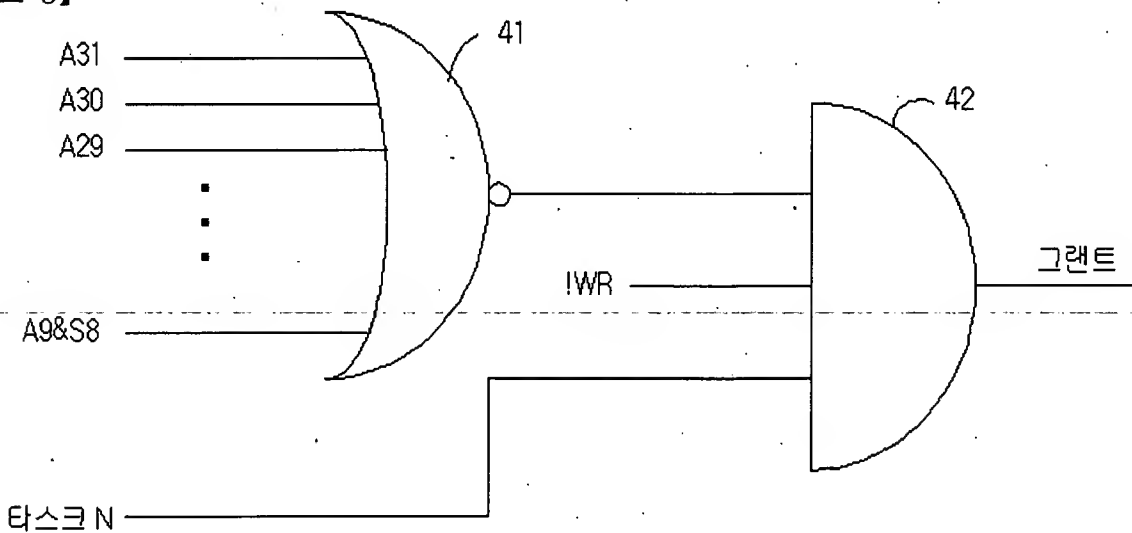
【도 1】



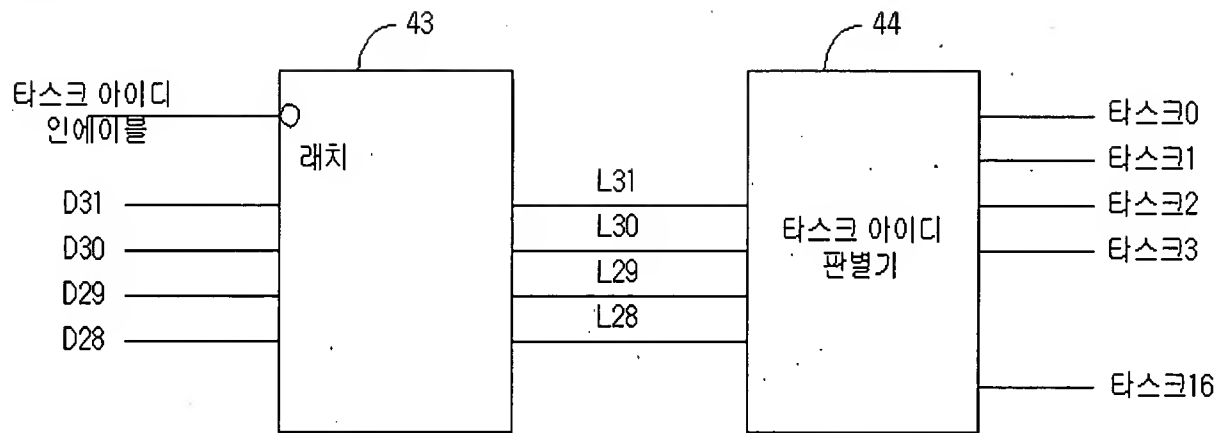
【도 2】



【도 3】



【도 4】



【도 5】

